Tool-supported continuous business process innovation: A case study in globally-distributed software teams

Alberto Heredia

Centro Universitario de Tecnología y Arte Digital (UTAD), Complejo Europa Empresarial, 28290 Las Rozas, Madrid, Spain Email: alberto.heredia@live.u-tad.com

Ricardo Colomo-Palacios

Faculty of Computer Sciences, Østfold University College, BRA Veien 4, 1783 Halden, Norway Email: ricardo.colomo-palacios@hiof.no

Pedro Soto-Acosta

Department of Management & Finance, University of Murcia, 30100 Espinardo, Murcia, Spain Email: psoto@um.es

Abstract

Software has a huge impact in modern society, so it is imperative to innovate continuously the way software products are created in order to ensure their high quality. During the execution of the various activities to produce software, individuals acquire tacit knowledge that can be useful to improve business processes. Even though people are geographically dispersed, social software supports the creation of knowledge clusters and provides additional channels to share knowledge for business process improvement. This paper describes a successful case study in which useful tacit knowledge is captured from a knowledge cluster with the aim of innovating services provided by a consultancy organisation. To this end, a knowledge-management-based framework helps to capture useful tacit knowledge, from individuals in different locations by using two social software tools during the production of software. These tools merge new tacit knowledge with existing organisational knowledge. Findings reveal that the use of this framework empowers the continuous innovation of business processes, thus allowing consultancy organisations to provide high-quality solutions. The different types of social software complemented one another as participants used each tool for a different purpose. Moreover, the framework allows newcomers to receive support from other colleagues and also mitigates the knowledge loss produced due to the high rotation of personnel in such organisations.

Keywords: continuous innovation, business processes, tacit knowledge, social software, global software development, knowledge management

1. Introduction

The impact of software in current society is deeper every year as more devices become part of our daily lives, e.g., smartphones, wearable devices, automated household appliances, and so on. To maintain a high level of quality in these products it is vital not only to improve the way they are created, but also to do this continuously (Ivarsson and Gorschek, 2012; Trkman, 2010). The complexity of these business processes in the software industry is constantly growing, so it is becoming capital for organisations to gain better insights into the way their business processes are executed in order to manage and improve them suitably (Delgado et al., 2014).

Within this context, software production relies intensively upon human capital because, in the end, software products are created by human beings, and hence their importance to the software industry is widely accepted (Colomo-Palacios et al., 2012). While individuals perform their different roles in the activities needed to produce software, they acquire tacit knowledge (Ivarsson and Gorschek, 2012). Since tacit knowledge is much larger than explicit knowledge, innovations in business processes should be based not just on the latter; useful experiences should be captured and made explicit (Del Giudice et al., 2013, 2015; Dingsøyr and Šmite, 2014; Palacios-Marqués et al., 2015). Knowledge clusters are important players in this difficult task. Enabled by established communities and empowered by new social networking tools, the wealth of knowledge networks is believed to foster a fertile environment for the exchange of knowledge, while also accelerating the innovation rate (Colomo-Palacios et al., 2014; Palacios-Marqués et al., 2015; Popa et al., 2016).

Nowadays, many software project teams are not located in one place but are spread out (Herbsleb and Moitra, 2001; Mishra and Mahanty, 2016). This brings some benefits for organisations, although it can also hamper the daily work of the teams, which may result in a reduction of the quality of software produced. Social software emerged as a solution to help mitigate these kinds of difficulties, as it can provide additional channels to share knowledge among distributed teams (Black et al., 2010).

Despite the importance of social software in supporting knowledge management activities, how it can be used to capture tacit knowledge and make it explicit to be shared in distributed environments for bringing innovations to software processes in the organisation is something that requires more attention from researchers.

This paper describes a successful case study using a framework designed to capture useful tacit knowledge that individuals in different geographic locations acquire in the execution of the business

processes and to merge it with the existing organisational knowledge. The main objective of the case study was to evaluate the benefits of using two social software tools within such a framework for the improvement of business processes. The research questions considered to fulfil this objective were:

RQ1. Does the use of two social software tools within the described framework help to break the barriers that inhibit individuals from sharing knowledge in GSD teams?

RQ2. Does the use of two social software tools within the described framework support newcomers in learning the appropriate knowledge to successfully adapt to the new environment?

RQ3. Does the use of two social software tools within the described framework help organisations to minimise the loss of knowledge due to personnel rotation?

RQ4. Does the use of two social software tools within the described framework allow organisation's management to keep control over processes?

RQ5. What is the level of quality of knowledge work using two social software tools within the described framework as seen from the different perspectives of the relevant groups for the business processes?

The ultimate aim of the framework is to empower the continuous improvement of business processes. To do so, social software is suggested as a means to ease the capture of tacit knowledge from individuals in different geographic locations and to make it explicit for the subsequent combination of this new knowledge with the existing organisational one. This framework was used in a case study to evaluate the convenience of its use in consultancy organisations, which strive to continuously improve the services they provide.

The remainder of the paper is organised as follows: Section 2 discusses related work for the management of knowledge in distributed teams to improve business processes in the software industry. Section 3 explains the framework for the use of social software tools. Following that, section 4 provides a description of the case study using that framework. Section 5 reports on the results of the case study, which are later discussed in Section 6. Different threats to the study conducted are analysed in section 7. The paper ends with the conclusions in Section 8.

2. Literature review

Business processes have become increasingly important in many organisations because they are the key drivers behind critical success factors –cost, quality and time– for developing value and distributing it to customers. These processes need to change frequently in response to business needs (Vera-Baquero et al., 2013), which implies that the organisation must systematically measure the way it carries them out to identify improvement opportunities and design changes to take advantage of those opportunities and implement them (Delgado et al., 2014). Once measures are implemented, it is critical to establish feedback loops to enable the continuous innovation in business processes. To do this, business process improvement lifecycle models are traditionally based on the well-known Deming Cycle (Walton, 1986) of Plan, Do, Check and Act (PDCA).The GQM$^+$ Strategies life cycle [(Basili et al., 2014), p.16] is one of these models that organisations usually tailor to their specific needs (Figure 1).

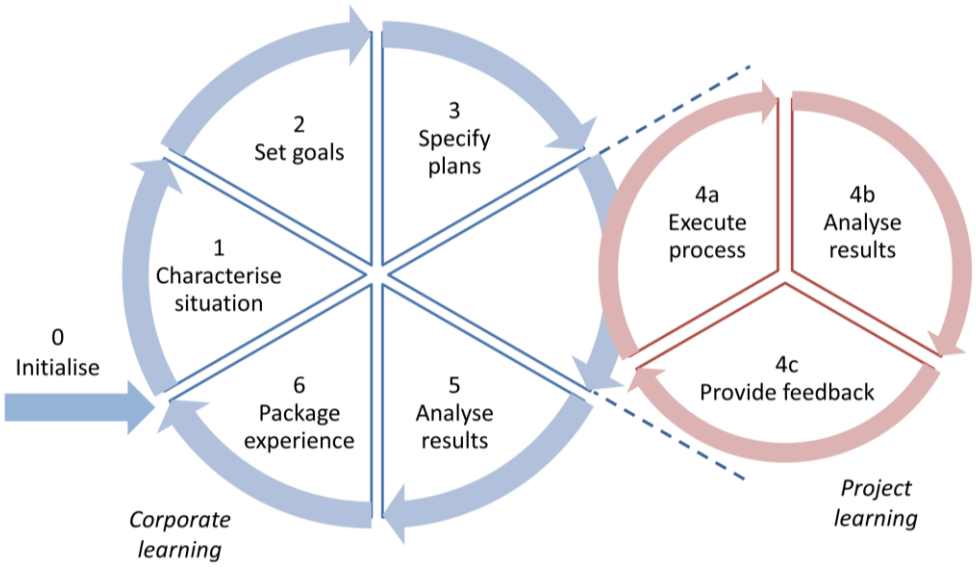

Figure 1 Basic GQM$^+$ Strategies life cycle, adapted from (Basili et al., 2014).

Business processes in industries like software production are often complex and variable because participants may execute a given task in several ways (Davenport, 2010). Thus, according to Seethamraju and Marjanovic (2009), they require improvement methods focused on Knowledge Management (KM) strategies and processes (Mazdeh and Hesamamiri, 2014; Mills and Smith, 2011), emphasizing the knowledge and experience that individual actors develop, use and share. Improvement initiatives should include, therefore, the participation of not only a few experts but also a variety of employees who are directly involved in these business processes (Del Giudice and

Maggioni, 2014; Marjanovic, 2013; Soto-Acosta et al., 2015). They have valuable tacit process knowledge acquired through experience that needs to be transformed into explicit knowledge (Nonaka et al., 2006) and shared within the organisation because they could become examples for other colleagues and create innovation opportunities (Soto-Acosta et al., 2014, 2015). As a result, the formation of social networks can have a positive impact on how work is done (Avram, 2007), so organisations should identify knowledge clusters and use social network techniques to support the successful management and improvement of their business processes (Colomo-Palacios et al., 2014; Palacios-Marqués et al., 2015; Popa et al., 2016).

However, it is not strange that people involved in a software project are located in multiple geographic locations, since Global Software Development (GSD) became a common practice in the software industry (Herbsleb and Moitra, 2001; Mishra and Mahanty, 2016). GSD brought some benefits, such as reduced development costs (Šmite et al., 2010), access to a larger and better-skilled workforce (Ebert, 2011) and a closer proximity to markets and customers (Conchúir et al., 2009). GSD also brought some challenges due to the dispersion of team members that have to deal with temporal, geographical and socio-cultural distances (Giuffrida and Dittrich, 2013). These difficulties usually result in communication problems, inefficient knowledge management and, eventually, a low-quality software production. To help overcome these issues and in addition to traditional communication channels, such as email, phone and video conferencing systems, social software offers an additional channel that can ultimately support and foster knowledge sharing in GSD teams (Black et al., 2010).

Previous works stressed the importance of social software to support KM activities for the improvement of business processes in the software industry through the interaction of human beings (Schmidt and Nurcan, 2009), although few studies focused on GSD, as highlighted by Giuffrida and Dittrich (2013).

The literature recently reported an analysis regarding the variety of challenges the global knowledge workers deal with when they apply social tools as a part of their working practices (Pirkkalainen and Pawlowski, 2014). Other authors have studied how communication and coordination tools were applied to support distributed teams (Portillo-Rodríguez et al., 2012) and proposed a theoretical framework for analysing how communicative practices through social software are constituted and maintained for coordination in globally-distributed software teams (Giuffrida and Dittrich, 2015). There are also studies on the positive and negative impacts of knowledge sharing platforms in

distributed software engineering processes, such as requirements (Ali and Lai, 2016; Sillaber and Breu, 2014) and a comparison of their use in both co-located and distributed teams (Gupta et al., 2009).

More specific examples are: the use of the social web and semantic annotations for electronic mentoring (Colomo-Palacios et al., 2014); the use of tags (Treude and Storey, 2009) to manage knowledge and innovate in software development processes within a distributed team environment; the study of the usefulness of wikis in corporate settings (Arazy et al., 2009) to collaboratively negotiate requirements with stakeholders (Wu et al., 2010) and enhance their quality through the management of experiences for organisational learning (Knauss et al., 2009), or to keep design documents updated and relevant using a wiki enhanced with a hierarchal glossary (Ben-Chaim et al., 2009).

Previous publications have studied how social software can provide useful channels to share knowledge in distributed software teams, but most of them focus mainly on the technological part, and few study the continuous innovation of business processes from the KM point of view. What is more, few studies have addressed more than one kind of social software (Giuffrida and Dittrich, 2013), so it is necessary to examine more than one communication channel and the interaction among these.

3. Framework Description

From the perspective of people as social beings, some specific issues have been previously reported by researchers with regard to the management of knowledge related to business processes in the software industry. In this work, we focus especially on:

1. The barriers that may inhibit individuals from sharing knowledge (Gavrilova and Andreeva, 2012), especially in GSD teams (Giuffrida and Dittrich, 2013), so hindering the transfer of knowledge at both project and organisation levels.
2. Newcomers to an organisation or a project team need to learn the appropriate knowledge to successfully adapt to the new environment (Beus et al., 2014).
3. Organisations have difficulties in minimising the loss of knowledge, due to personnel rotation (Siegel Christian et al., 2014).

4. Organisation's management considers it a risk to allow knowledge contributions from individuals throughout the organisation to improve the business processes because they may lose control over processes by allowing increased collaboration (Kemsley, 2010).
5. It is challenging to measure the quality of knowledge work as it usually depends on the context (Davenport, 2010).

The framework described in this section addresses these issues with the support of social software. It aims to capture the tacit knowledge that rises during the business process execution and merge it with the existing explicit knowledge in the organisation to empower the continuous improvement of business processes. This KM-based collaborative framework, adapted from (Heredia et al., 2013), holds that everyone involved in the business processes, no matter what the location, should be able to provide ideas to improve them when an opportunity is found.

Within this framework, knowledge flows according to the GQM$^+$ Strategies life cycle represented in Figure 1, so a parallelism can be made. As Figure 2 shows, the current business processes of the organisation are identified and formalised prior to the continuous cycle (phase 0), i.e., they are defined in the form of explicit knowledge and stored in a shared organisational repository. This phase is optional as the framework is still useful even without an initial repository of business processes, which is often missing or incomplete; an empty organisational repository would be populated at the end of the cycle with the knowledge gathered in the projects.

When a new project starts, people in the work team should have clear guidance about the processes they have to execute, so process experts analyse which organisational knowledge is required by the team according to the project needs (phase 1). Then, the librarian who manages the organisational repository selects those knowledge assets that could guide each team in their projects (phase 2) and spreads them to the social software used by the corresponding distributed software teams (phase 3). For instance, the librarian could select guides, documents, examples and lessons learned stored in the organisational repository about a process that a team has to execute in a project and would publish them in a wiki or a microblog to make it available for the team members.

When needed during the project execution, the social software will allow team members to easily access and reuse knowledge related to a specific business process or to a specific role (phase 4a). The experience gained by individuals during the project is internalised so that they acquire new tacit knowledge (phase 4b). If this tacit knowledge leads anyone in the team to discover an opportunity for business process improvement, they can use the social software to provide feedback about the

experience (phase 4c). The social software allows these contributions to enhance the quality of the business process and the results are visible to all people involved in the project; without social software, team members would use other communication channels (e.g. e-mail) to share knowledge that would not be captured, hence missing out on potential improvement opportunities.

This framework considers not only the possibility of sharing knowledge within the scope of a project, but also at the organisational level. To do so, new knowledge captured from distributed software teams working in the different projects of the organisation can be gathered (phase 5). A supervision mechanism allows process experts to verify if new contributions are aligned with the business strategy so that they can be merged with the organisational knowledge (phase 6).

Thus, these six phases clearly define two feedback loops: the first occurs within the fourth phase, when the business processes are executed, and represents the learning at project level; the second comprises the whole cycle and represents learning at organisational level.

Once the organisational knowledge has been upgraded, the process experts should analyse if new contributions could be useful for other projects in the organisation and need to be spread in consequence, thus starting the cycle again.

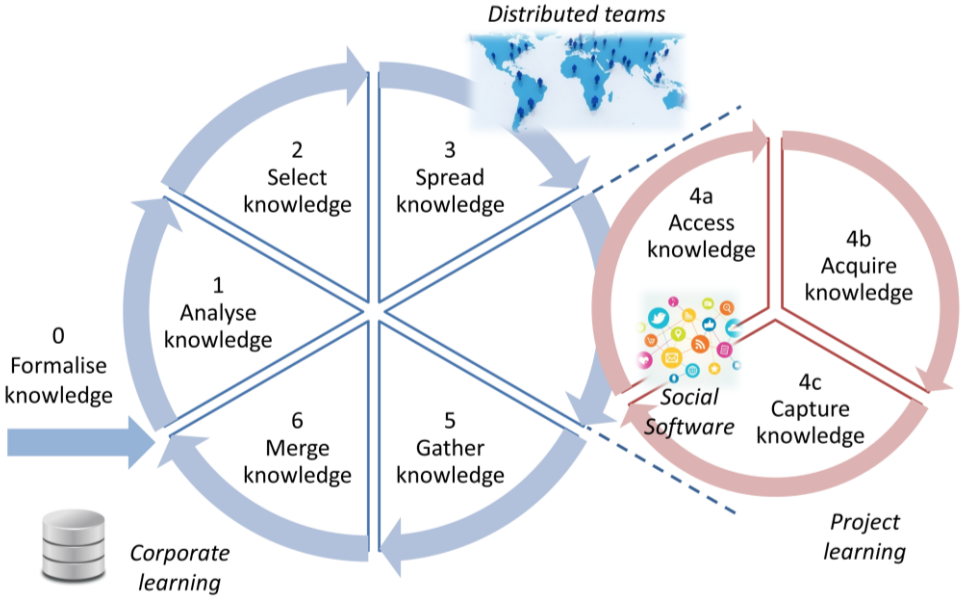

Figure 2 Knowledge life cycle within the proposed framework.

4. Case Study

The objective of the case study carried out in the scope of this research work was to evaluate the benefits of using two social software tools within the described framework for the improvement of

business processes. The case study took place in an agile software development environment, where engineers worked in teams spread in different locations to deliver different software products meeting requirements established by the corresponding customers. Within this environment, new knowledge contributed by engineers during the execution of the business processes was supervised by process experts to verify if it was aligned with the business strategy.

As knowledge work depends on the context, it is possible to measure its quality subjectively by asking relevant people about the particular work involved. Therefore, not only individuals executing the business processes but also other stakeholders (process experts and customers) were considered for participation in the case study.

The research questions considered to fulfil the objective of the case study were:

RQ1. Does the use of two social software tools within the described framework help to break the barriers that inhibit individuals from sharing knowledge in GSD teams?

RQ2. Does the use of two social software tools within the described framework support newcomers in learning the appropriate knowledge to successfully adapt to the new environment?

RQ3. Does the use of two social software tools within the described framework help organisations to minimise the loss of knowledge due to personnel rotation?

RQ4. Does the use of two social software tools within the described framework allow organisation's management to keep control over processes?

RQ5. What is the level of quality of knowledge work using two social software tools within the described framework as seen from the different perspectives of the relevant groups for the business processes?

The case study was conducted over the course of several years. By the end of the collection of data in 2014, it comprised a sample of 228 individuals distributed in 4 locations with ages ranging from 22 to 39 years. Participants were selected from those who responded positively to a personal invitation sent by the authors to professionals tied in to the Spanish consulting industry. The sample included 36 women (15.8 %) and 192 men (84.2 %), which is a gender imbalance typically present in the computer science population. Fourteen participants were process experts and another fourteen were customers. A subset of 64 participants (32 %) worked in teams to produce software within

environments based on the framework depicted in Figure 2 and had the support of wikis and microblogs to manage the knowledge involved in the software production. The remaining 136 participants (68 %) worked in teams to produce software without the support of any social software.

At the end of the corresponding projects, printed questionnaires were administered to participants, who were also assisted on site by a researcher –when needed– with instructions on how to fill out the questionnaire. Subsequently, responses were codified using a statistical analysis software tool.

Regarding the indicators used in the case study, on the one hand we measured the volume of knowledge produced during the projects when using the proposed framework and the ease of the mechanisms deployed to contribute that knowledge. On the other, quality should not be limited to one generalised construct, because software products and services have many dimensions; hence, quality of the knowledge work and its impact were assessed in this research by considering different perspectives: quality of the software products delivered in terms of meeting customer expectations, quality of the processes to produce software as seen by process experts, and quality of the knowledge assets managed during projects from the point of view of participants.

5. Results

The analysis of the information gathered in the scope of the case study shows that the use of two social software tools within the framework depicted in Figure 2 addresses the issues presented in the previous section in the manner below:

RQ1. The use of two social software tools provides individuals in GSD teams with different additional channels to quickly share new knowledge that comes from experience with little effort; while microblogs provide immediacy and entail little effort, wikis can provide richer knowledge. In addition, the two feedback loops considered in the proposed framework allow this knowledge to be shared at both project and organisation level. Respondents stated that the mechanisms to contribute new knowledge to the wikis were not difficult to use in general (rated 0.61 on average on a [0..1] scale), although contributing examples of work products required more effort on their part to provide some context about the example; respondents also pointed out the ease and immediacy of contributing new knowledge using microblogs (rated 0.79 on average on a [0..1] scale) as they only require a short text. None of the participants reported a negative influence between the wiki and the microblog.

RQ2. The use of social software also triggers mechanisms of sociality, so it provides newcomers with a forum of interaction to receive support from other colleagues. Furthermore, knowledge was made explicit before they became part of the organisation or the project became available, so they can learn from previous experiences to successfully adapt to the new environment. The volume of new knowledge produced during the case study is considered high; participants contributed to the wikis with more than 500 examples of work products, participated 78 times in threaded discussions attached to different knowledge assets, and updated 21 existing knowledge assets; a total of 134 tweets were also sent during the case study. However, respondents agreed that the use of social software does not promote, by itself, the creation of a community of users of the business processes; it has to be individuals who create such a community with their contributions.

RQ3. Participants highlighted the fact that experiences of former team members were not lost because these experiences were really useful for their learning of the software processes, as was mentioned before. When a team member shared new tacit knowledge using the wikis or the microblogs, it was then made explicit and eventually became part of the organisational knowledge. Thus when a project team lost a member, the knowledge loss was minimised.

RQ4. The process experts declared in the questionnaires that the supervision mechanism considered in the proposed framework assures that accepted contributions are aligned with the business strategy, thus allowing organisation's management to maintain control over processes.

RQ5. Quality of the knowledge work using the proposed framework and its impact were assessed from different perspectives (individuals executing the business processes, process experts and customers). Findings reveal that the use of social software can: (1) increase the quality of software products in terms of meeting customer expectations, (2) enhance the quality of the processes to produce software as seen by process experts, and (3) maintain a good level of quality in the knowledge assets managed during projects from the point of view of participants. The analysis of results leading to these conclusions is detailed below.

Table 1 summarises the distribution of the data collected during the case study about the quality of the software products delivered in terms of meeting customer expectations, the quality of the processes to produce software as seen by process experts, and the quality of the knowledge

managed during projects from the point of view of participants. These data are classified depending on the Social Software (SoSo) used, i.e. none, wiki and microblog. All the measures were ranked on a [0..1] scale, where 0 means poor quality and 1 means excellent quality. The quality of knowledge was only evaluated when a Social Software was used to manage it.

Table 1 Mean and standard deviation for the measures considered in the case study.

SoSo	Teams	Quality of products		Quality of processes		Quality of knowledge	
		Mean	Std Dev	Mean	Std Dev	Mean	Std Dev
None	37	0.7901	0.1870	0.6953	0.1548	N/A	N/A
Wiki	12	0.8889	0.0978	0.7157	0.1133	0.6213	0.2579
Microblog	7	0.7902	0.2253	0.8023	0.0898	0.6404	0.2939

Quality of the software products delivered

With regard to the quality of the software products delivered in terms of meeting customer expectations, Table 1 reveals a higher mean value when social software is used as a means to manage the knowledge involved in the software production, especially in the case of wikis.

Table 2 ANOVA results for testing the difference in the quality of the software products delivered.

	Df	Sum Sq	Mean Sq	F value	Pr(>F)
SoSo	2	0.0914	0.04571	1.498	0.233
Residuals	53	1.6173	0.03051		

Table 2 presents the results of a further study of these data. Results are, however, not conclusive because the significant difference is not lower than 0.05.

Levene's test for equality of the population variances shows homoscedasticity (p-value = 0.1476), so a multiple comparison using Tukey's 'Honest Significant Difference' method can be performed using a family-wise confidence level of 95 % (Table 3).

Table 3 Tukey's HSD results for comparing the differences in the quality of the software products delivered.

	Diff	Lwr	Upr	p adj
Microblog-None	0.04598610	-0.12762287	0.2195951	0.7995035
Wiki-None	0.09883491	-0.04109233	0.2387622	0.2134965
Wiki-Microblog	0.05284881	-0.14747514	0.2531728	0.8009352

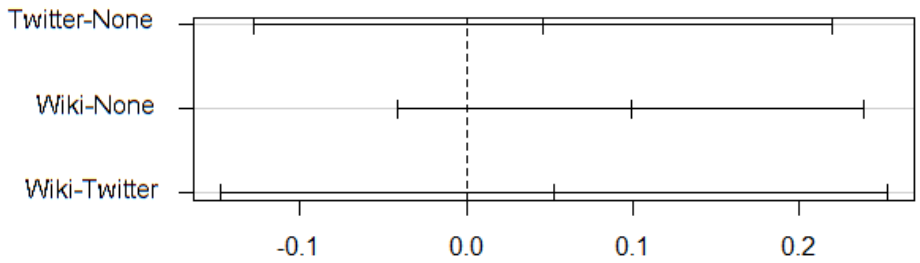

Figure 3 Tukey's HSD results for comparing the differences in the quality of the software products delivered.

Representation of results (Figure 3) confirms that the use of social software can increase the quality of the software products delivered in terms of meeting customer expectations. The increase was higher in teams that used a wiki, although results are not statistically significant (p-value = 0.2134965). This difference between using wikis or microblogs may be due to the fact that participants contributed to the knowledge in the wikis with many examples of work products, which led other colleagues to improve other products and hence customer satisfaction; in contrast, the maximum number of characters allowed in microblog messages meant a limitation in this sense.

Quality of the processes to produce software

Regarding the quality of the processes to produce software as seen by process experts, Table 1 also shows a higher mean value when social software is used as a means to manage the knowledge involved in the software production, especially in the case of microblogs.

Table 4 ANOVA results for testing the difference in the quality of the processes to produce software.

	Df	Sum Sq	Mean Sq	F value	Pr(>F)
SoSo	2	0.0675	0.03375	1.7	0.193
Residuals	53	1.0525	0.01986		

Table 4 presents the results of a further study of these data. Results are, however, not conclusive because the significant difference is not lower than 0.05.

Levene's test for equality of the population variances shows homoscedasticity (p-value = 0.1853), so a multiple comparison using Tukey's 'Honest Significant Difference' method can be performed using a family-wise confidence level of 95 % (Table 5).

Table 5 Tukey's HSD results for comparing the differences in the quality of the processes to produce software.

	Diff	lwr	upr	p adj
Microblog-None	0.10700039	-0.03305263	0.24705340	0.1659084
Wiki-None	0.02048491	-0.09239654	0.13336636	0.9000434
Wiki-Microblog	-0.08651548	-0.24811987	0.07508892	0.4065365

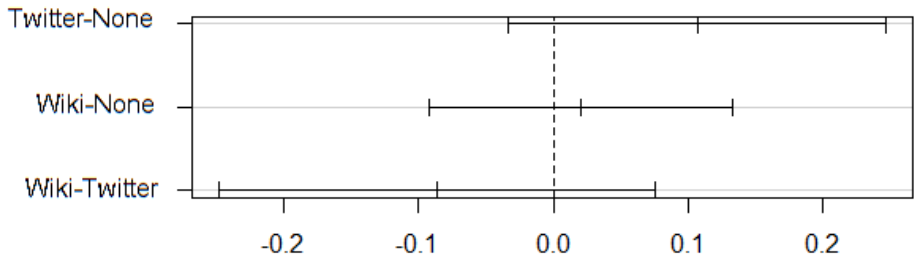

Figure 4 Tukey's HSD results for comparing the differences in the quality of the processes to produce software.

Results depicted in Figure 4 confirm that the use of social software can increase the quality of the processes to produce software as seen by process experts. The increase was especially significant in teams that used microblogs, although results are not statistically significant (p-value = 0.1659084). This difference between using microblogs or wikis may be due to the fact that microblogs were used to remind participants about key aspects of each process and to provide a mechanism to solve doubts, so it led them to better performances; in contrast, the wikis provided more detailed (and dense) knowledge about the business processes.

Quality of the knowledge assets managed during projects

From the point of view of the individuals involved in the execution of business processes, Table 1 shows a good level in the quality of the knowledge assets managed during projects using both the wikis and microblogs. The mean was slightly higher when microblogs were used, but the Student's t-test performed with a confidence level of the interval equal to 0.95 does not allow us to accept the hypothesis that the quality is higher when microblogs are used (p-value = 0.3504). The supervision mechanism considered in the proposed framework allows process experts to establish the level of quality that is required for new contributions to be accepted, thus assuring the high quality of the organisational knowledge about the business processes.

6. Discussion

The following paragraphs present a comparison of the findings in this research work with others already reported in the literature.

According to the results of the case study, social software provides individuals with an additional channel to share new knowledge that comes from experience quickly and easily. This finding is in line with research by (Erol et al., 2010), who concluded that the lack of formal barriers in social software also tears down psychological barriers, so anyone can contribute to the improvement of business processes without excuses.

Respondents in our case study also highlighted some differences depending on the social software used: microblogs provide immediacy and entail little effort to share knowledge, given that they only require a short text, which also represents a limitation; in contrast, wikis allow people to contribute richer knowledge (e.g. examples of work products), although they sometimes require a bit more effort (e.g. to provide some context about those examples). These opinions from our respondents regarding both communication channels are in contrast with the work of Giuffrida and Dittrich (2013).

Our participants confirmed that social software encourages sociality of team members and integrates individuals into the business process life cycle, a benefit already stated in the literature (Bruno et al., 2011). Social software consequently helps to avoid the sense of isolation that newcomers especially may feel when they join a new project team. Rotation of personnel could mean newcomers receive less support in personnel integration because they are expected to remain in the team for a short time (Colomo-Palacios et al., 2011). However, our results confirm that social software provides them with a forum of interaction to receive support from other colleagues and to learn from previously shared knowledge, since prior experiences can play an important role in helping newcomers to adjust their performance to a new work context (Beus et al., 2014). The volume of new knowledge produced during the case study was considered high, which confirms conclusions of other authors regarding the improvement in the exchange of knowledge (Erol et al., 2010) and the increase in the involvement and commitment of employees (Prilla and Nolte, 2012). Nevertheless, respondents agreed that the use of social software does not promote, in itself, the creation of a community of users of the business processes; it has to be people who create this community with their contributions, a finding aligned with recent research (Bruno et al., 2011) that

emphasises the fact that people are motivated to participate using their social identities only if they feel part of the community.

In addition to the promotion of sociality, making this knowledge explicit allows organisations to minimise the knowledge loss if a project team changes a member. Some authors had already pointed out the vulnerability of companies to the erosion of their organisational knowledge through the leaving of key employees (O'Connor and Basri, 2014).Project teams do no easily adapt when a member is removed from the team (Siegel Christian et al., 2014) and the loss of knowledge is often not recognized until too late, but our proposal anticipates this situation because experiences of employees would have been previously made explicit and would have become part of the organisational knowledge.

On the one hand, our findings show that allowing knowledge contributions from individuals throughout the organisation does not necessarily imply that the organisation's management loses control over processes, as claimed by (Kemsley, 2010). This risk is reduced thanks to the supervision mechanism included in the proposed framework, which allows the *a posteriori* control of quality (Bruno et al., 2011).The use of this mechanism provides results that complement previous research (Prilla and Nolte, 2012). Although expert intervention might discourage people from active participation, the responsibility of process experts in our framework is not to guide people but to ensure that accepted contributions are aligned with the business strategy. This importance of process experts to ensure the quality of the organisational knowledge is also pointed out by Prilla and Nolte (2012). On the other hand, in contrast to the conclusions of (Schmidt and Nurcan, 2009), where the authors stress the lack of hierarchy of social software as a disadvantage, the use of this mechanism also introduces a hierarchy in the management of knowledge, because process experts have the power to decide whether some knowledge that was first spread at project level can be later spread at organisation level too.

Finally, given the difficulties of properly measuring knowledge work, we followed the recommendations of (Davenport, 2010) and used a subjective method involving the relevant groups for the particular business processes in our case study and asked them for their thoughts. Their answers revealed that the use of social software can increase the quality of the knowledge work considering three different perspectives. First, social software enhances the quality of software products in terms of meeting customer expectations, a finding that is in line with the article by von

Krogh (2012) and provides a different point of view than the work by Culnan et al. (2010), who analysed the use of social software to interact with customers, rather than for internal knowledge sharing. Second, social software increases the quality of the processes to produce software as seen by process experts. This confirms other research (Koschmider et al., 2010) in which the use of social software helps to collaboratively enhance a repository of high-quality process models. Third, social software maintains a good level of quality in the knowledge assets managed during projects from the point of view of the individuals executing the business processes; not only does the supervision mechanism assure a certain level of quality, we also have to take into account that deficits in quality damage the reputation of the individual who contributed the knowledge, so people will do their best to achieve a maximum degree of quality (Bruno et al., 2011).

7. Threats to validity

The purpose of this section is to analyse the different threats to the study conducted regarding construct validity, internal validity, external validity and conclusion validity. Countermeasures taken against these threats to validity are also mentioned.

Construct validity refers to the accuracy of the research strategy and the variables considered for discussing the research questions. To minimise these threats to validity, the research plan and the measures to be taken were discussed and refined with other researchers. In addition, researchers assisted participants in filling out the questionnaire in order to avoid different interpretations of the questions.

Internal validity is of concern when causal relations are examined. There is a risk that our results were not due to the use of social software for managing knowledge related to business processes but to another factor. Researchers consider that the variety of projects involved in the case study was enough to reduce their influence in results. Additionally, researchers confirmed that all participants had comparable levels of knowledge and experience in software development.

Concerning external validity, which is related to the generalisability of research findings, the authors assume two possible threats. These are the limited number of participants in the case study and the representativeness of the sample. However, due to the nature of the study, the authors consider that the sample had an acceptable statistical power to generalise the findings.

Finally, regarding conclusion validity, the authors paid much attention to providing statistically-correct conclusions based on the data gathered. Yet most of the measures were obtained through

questionnaires, so they represent the subjective point of view of participants. The anonymity of the questionnaires helped to reduce this threat by allowing participants to provide honest answers.

8. Conclusions

Human capital is vital to continuously refine business processes because individuals acquire tacit knowledge during the execution of those processes that could be useful to upgrade them. This paper presented a successful case study in which two social software tools were used following a KM-based collaborative framework; this framework is aimed at capturing useful tacit knowledge from individuals through social software during the production of software and merging it with the existing organisational knowledge with the intention of improving the business processes. The case study carried out validated the support of this framework in the continuous improvement of the services provided by consultancy organisations. Findings confirmed that the use of social software can help people to deal with some specific issues reported by researchers with regard to the management of knowledge related to business processes in the software industry. First, social software provides additional channels for sharing new knowledge that comes from experience among individuals in different locations, at both project and organisation level; while microblogs provide immediacy and entail little effort, wikis can provide richer knowledge, so both tools can coexist without negative influences. The use of social software also triggers mechanisms of sociality, so it provides newcomers with a forum of interaction to receive support from other colleagues and to learn from shared experiences. In addition, making this knowledge explicit allows organisations to minimise the knowledge loss if a project team loses a member. On the other hand, allowing knowledge contributions from individuals throughout the organisation does not necessarily imply a loss of control over processes thanks to the supervision mechanism included in the proposed framework, which assures that accepted contributions are aligned with the business strategy. Finally, social software has also proven to increase the quality of software products in terms of meeting customer expectations, enhance the quality of the processes to produce software as seen by process experts, and maintain a good level of quality in the knowledge assets managed during projects from the point of view of the people executing the business processes.

Acknowledgments

We would like to thank Fundación CajaMurcia for the support provided.

References

Ali, N. and Lai, R. (2016) 'A method of requirements change management for global software development', *Information and Software Technology*, Vol. 70, pp. 49–67. DOI: 10.1016/j.infsof.2015.09.005.

Arazy, O., Gellatly, I., Jang, S. and Patterson, R. (2009) 'Wiki deployment in corporate settings', *IEEE Technology and Society Magazine*, Vol. 28, No. 2, pp. 57–64. DOI: 10.1109/MTS.2009.932804.

Avram, G. (2007) 'Of Deadlocks and Peopleware - Collaborative Work Practices in Global Software Development', in *Proceedings of the 2nd International Conference on Global Software Engineering*. IEEE, pp. 91–102. DOI: 10.1109/ICGSE.2007.30.

Basili, V., Trendowicz, A., Kowalczyk, M., Heidrich, J., Seaman, C., Münch, J. and Rombach, D. (2014) *Aligning Organizations Through Measurement: The GQM+ Strategies Approach*. Springer, Kaiserslautern, Germany.

Ben-Chaim, Y., Farchi, E. and Raz, O. (2009) 'An effective method for keeping design artifacts up-to-date', in *Proceedings of the 4th ICSE Workshop on Wikis for Software Engineering*, pp. 1–6. DOI: 10.1109/WIKIS4SE.2009.5069991.

Beus, J. M., Jarrett, S. M., Taylor, A. B. and Wiese, C. W. (2014) 'Adjusting to new work teams: Testing work experience as a multidimensional resource for newcomers', *Journal of Organizational Behavior*, Vol. 35, No. 4, pp. 489–506. DOI: 10.1002/job.1903.

Black, S., Harrison, R. and Baldwin, M. (2010) 'A Survey of Social Media Use in Software Systems Development', in *Proceedings of the 1st Workshop on Web 2.0 for Software Engineering*. ACM, pp. 1–5. DOI: 10.1145/1809198.1809200.

Bruno, G., Dengler, F., Jennings, B., Khalaf, R., Nurcan, S., Prilla, M., Sarini, M., Schmidt, R. and Silva, R. (2011) 'Key challenges for enabling agile BPM with social software', *Journal of Software Maintenance and Evolution: Research and Practice*, Vol. 23, No. 4, pp. 297–326. DOI: 10.1002/smr.523.

Colomo-Palacios, R., Casado-Lumbreras, C., Soto-Acosta, P. and Misra, S. (2014) 'Providing knowledge recommendations: an approach for informal electronic mentoring', *Interactive Learning Environments*, Vol. 22, No. 2, pp. 221–240. DOI: 10.1080/10494820.2012.745430.

Colomo-Palacios, R., Casado-Lumbreras, C., Soto-Acosta, P., Misra, S. and García-Peñalvo, F. J. (2012) 'Analyzing Human Resource Management Practices Within the GSD Context', *Journal of Global Information Technology Management*, Vol. 15, No. 3, pp. 30–54. DOI: 10.1080/1097198X.2012.10845617.

Colomo-Palacios, R., Fernandes, E., Soto-Acosta, P. and Sabbagh, M. (2011) 'Software product evolution for Intellectual Capital Management: The case of Meta4 PeopleNet', *International*

Journal of Information Management, Vol. 31, No. 4, pp. 395–399. DOI: 10.1016/j.ijinfomgt.2011.04.001.

Colomo-Palacios, R., Messnarz, R., Siakas, K., Palosi, D. and Coakley, D. (2014) 'Using social media as a tool for business improvement and certification of knowledge workers', *Journal of Software: Evolution and Process*, Vol. 26, No. 9, pp. 791–798. DOI: 10.1002/smr.1668.

Conchúir, E. Ó., Ågerfalk, P. J., Olsson, H. H. and Fitzgerald, B. (2009) 'Global Software Development: Where Are the Benefits?', *Communications of the ACM*, Vol. 52, No. 8, pp. 127–131. DOI: 10.1145/1536616.1536648.

Culnan, M. J., McHugh, P. J. and Zubillaga, J. I. (2010) 'How Large U.S. Companies Can Use Twitter and Other Social Media to Gain Business Value', *MIS Quarterly Executive*, Vol. 9, No. 4, pp. 243–259.

Davenport, T. H. (2010) 'Process Management for Knowledge Work', in Brocke, P. D. J. vom and Rosemann, P. D. M. (Eds.) *Handbook on Business Process Management 1*. Springer Berlin Heidelberg, pp. 17–35.

Delgado, A., Weber, B., Ruiz, F., Garcia-Rodríguez de Guzmán, I. and Piattini, M. (2014) 'An integrated approach based on execution measures for the continuous improvement of business processes realized by services', *Information and Software Technology*, Vol. 56, No. 2, pp. 134–162. DOI: 10.1016/j.infsof.2013.08.003.

Dingsøyr, T. and Šmite, D. (2014) 'Managing Knowledge in Global Software Development Projects', *IT Professional*, Vol. 16, No. 1, pp. 22–29. DOI: 10.1109/MITP.2013.19.

Ebert, C. (2011) *Global Software and IT: A Guide to Distributed Development, Projects, and Outsourcing*. John Wiley & Sons, Hoboken, NJ, USA.

Erol, S., Granitzer, M., Happ, S., Jantunen, S., Jennings, B., Johannesson, P., Koschmider, A., Nurcan, S., Rossi, D. and Schmidt, R. (2010) 'Combining BPM and social software: contradiction or chance?', *Journal of Software Maintenance and Evolution: Research and Practice*, Vol. 22, No. 6-7, pp. 449–476. DOI: 10.1002/smr.460.

Gavrilova, T. and Andreeva, T. (2012) 'Knowledge elicitation techniques in a knowledge management context', *Journal of Knowledge Management*, Vol. 16, No. 4, pp. 523–537. DOI: 10.1108/13673271211246112.

Del Giudice, M. and Maggioni, V. (2014) 'Managerial practices and operative directions of knowledge management within inter-firm networks: a global view', *Journal of Knowledge Management*, Vol. 18, No. 5, pp. 841–846. DOI: 10.1108/JKM-06-2014-0264.

Del Giudice, M., Peruta, M. R. Della and Maggioni, V. (2013) 'Collective Knowledge and Organizational Routines within Academic Communities of Practice: an Empirical Research on Science–Entrepreneurs', *Journal of the Knowledge Economy*, Vol. 4, No. 3, pp. 260–278. DOI: 10.1007/s13132-013-0158-3.

Del Giudice, M., Peruta, M. R. Della and Maggioni, V. (2015) 'A model for the diffusion of knowledge sharing technologies inside private transport companies', *Journal of Knowledge Management*, Vol. 19, No. 3, pp. 611–625. DOI: 10.1108/JKM-02-2015-0047.

Giuffrida, R. and Dittrich, Y. (2013) 'Empirical studies on the use of social software in global software development – A systematic mapping study', *Information and Software Technology*, Vol. 55, No. 7, pp. 1143–1164. DOI: 10.1016/j.infsof.2013.01.004.

Giuffrida, R. and Dittrich, Y. (2015) 'A conceptual framework to study the role of communication through social software for coordination in globally-distributed software teams', *Information and Software Technology*, Vol. 63, pp. 11–30. DOI: 10.1016/j.infsof.2015.02.013.

Gupta, A., Mattarelli, E., Seshasai, S. and Broschak, J. (2009) 'Use of collaborative technologies and knowledge sharing in co-located and distributed teams: Towards the 24-h knowledge factory', *The Journal of Strategic Information Systems*, Vol. 18, No. 3, pp. 147–161. DOI: 10.1016/j.jsis.2009.07.001.

Herbsleb, J. D. and Moitra, D. (2001) 'Global software development', *IEEE Software*, Vol. 18, No. 2, pp. 16–20. DOI: 10.1109/52.914732.

Heredia, A., Guzmán, J. G., Amescua, A. and Segura, M. I. S. (2013) 'Interactive Knowledge Asset Management: Acquiring and Disseminating Tacit Knowledge', *Journal of Information Science and Engineering*, Vol. 29, No. 1, pp. 133–147.

Ivarsson, M. and Gorschek, T. (2012) 'Tool support for disseminating and improving development practices', *Software Quality Journal*, Vol. 20, No. 1, pp. 173–199. DOI: 10.1007/s11219-011-9139-6.

Kemsley, S. (2010) 'Enterprise 2.0 Meets Business Process Management', in Brocke, P. D. J. vom and Rosemann, P. D. M. (Eds.) *Handbook on Business Process Management 1*. Springer Berlin Heidelberg, pp. 565–574.

Knauss, E., Brill, O., Kitzmann, I. and Flohr, T. (2009) 'SmartWiki: Support for high-quality requirements engineering in a collaborative setting', in *Proceedings of the 4th ICSE Workshop on Wikis for Software Engineering*, pp. 25–35. DOI: 10.1109/WIKIS4SE.2009.5069994.

Koschmider, A., Song, M. and Reijers, H. A. (2010) 'Social software for business process modeling', *Journal of Information Technology*, Vol. 25, No. 3, pp. 308–322. DOI: 10.1057/jit.2009.21.

von Krogh, G. (2012) 'How does social software change knowledge management? Toward a strategic research agenda', *The Journal of Strategic Information Systems*, Vol. 21, No. 2, pp. 154–164. DOI: 10.1016/j.jsis.2012.04.003.

Marjanovic, O. (2013) 'Knowledge-Intensive Healthcare Processes: Rethinking Business Process Ownership', in *Proceedings of the 46th Hawaii International Conference on System Sciences*. IEEE Computer Society, pp. 3416–3425. DOI: 10.1109/HICSS.2013.378.

Mazdeh, M. M. and Hesamamiri, R. (2014) ‘Knowledge management reliability and its impact on organizational performance: An empirical study’, *Program: electronic library & information systems*, Vol. 48, No. 2, pp. 102–126. DOI: 10.1108/PROG-01-2013-0001.

Mills, A. M. and Smith, T. A. (2011) ‘Knowledge management and organizational performance: a decomposed view’, *Journal of Knowledge Management*, Vol. 15, No. 1, pp. 156–171. DOI: 10.1108/13673271111108756.

Mishra, D. and Mahanty, B. (2016) ‘A study of software development project cost, schedule and quality by outsourcing to low cost destination’, *Journal of Enterprise Information Management*, Vol. 29, No. 3, pp. 454–478. DOI: 10.1108/JEIM-08-2014-0080.

Nonaka, I., von Krogh, G. and Voelpel, S. (2006) ‘Organizational Knowledge Creation Theory: Evolutionary Paths and Future Advances’, *Organization Studies*, Vol. 27, No. 8, pp. 1179–1208. DOI: 10.1177/0170840606066312.

O’Connor, R. V. and Basri, S. (2014) ‘Understanding the Role of Knowledge Management in Software Development:: A Case Study in Very Small Companies’, *International Journal of Systems and Service-Oriented Engineering*, Vol. 4, No. 1, pp. 39–52. DOI: 10.4018/ijssoe.2014010103.

Palacios-Marqués, D., Merigó, J. M. and Soto-Acosta, P. (2015) ‘Online social networks as an enabler of innovation in organizations’, *Management Decision*, Vol. 53, No. 9, pp. 1906–1920. DOI: 10.1108/MD-06-2014-0406.

Palacios-Marqués, D., Soto-Acosta, P. and Merigó, J. M. (2015) ‘Analyzing the effects of technological, organizational and competition factors on Web knowledge exchange in SMEs’, *Telematics and Informatics*, Vol. 32, No. 1, pp. 23–32. DOI: 10.1016/j.tele.2014.08.003.

Pirkkalainen, H. and Pawlowski, J. M. (2014) ‘Global social knowledge management – Understanding barriers for global workers utilizing social software’, *Computers in Human Behavior*, Vol. 30, pp. 637–647. DOI: 10.1016/j.chb.2013.07.041.

Popa, S., Soto-Acosta, P. and Loukis, E. (2016) ‘Analyzing the complementarity of web infrastructure and eInnovation for business value generation’, *Program*, Vol. 50, No. 1, pp. 118–134. DOI: 10.1108/PROG-09-2015-0065.

Portillo-Rodríguez, J., Vizcaíno, A., Piattini, M. and Beecham, S. (2012) ‘Tools used in Global Software Engineering: A systematic mapping review’, *Information and Software Technology*, Vol. 54, No. 7, pp. 663–685. DOI: 10.1016/j.infsof.2012.02.006.

Prilla, M. and Nolte, A. (2012) ‘Integrating Ordinary Users into Process Management: Towards Implementing Bottom-Up, People-Centric BPM’, in Bider, I., Halpin, T., Krogstie, J., Nurcan, S., Proper, E., Schmidt, R., Soffer, P., and Wrycza, S. (Eds.) *Enterprise, Business-Process and Information Systems Modeling*. Springer Berlin Heidelberg, pp. 182–194.

Schmidt, R. and Nurcan, S. (2009) ‘BPM and Social Software’, in Ardagna, D., Mecella, M., and Yang, J. (Eds.) *Business Process Management Workshops*. Springer Berlin Heidelberg, pp. 649–658.

Seethamraju, R. and Marjanovic, O. (2009) 'Role of process knowledge in business process improvement methodology: a case study', *Business Process Management Journal*, Vol. 15, No. 6, pp. 920–936. DOI: 10.1108/14637150911003784.

Siegel Christian, J., Pearsall, M. J., Christian, M. S. and Ellis, A. P. J. (2014) 'Exploring the benefits and boundaries of transactive memory systems in adapting to team member loss', *Group Dynamics: Theory, Research, and Practice*, Vol. 18, No. 1, pp. 69–86. DOI: 10.1037/a0035161.

Sillaber, C. and Breu, R. (2014) 'The Impact of Knowledge Sharing Platforms in Distributed Requirements Engineering Scenarios: A Systematic Review', in Uden, L., Wang, L. S. L., Rodríguez, J. M. C., Yang, H.-C., and Ting, I.-H. (Eds.) *The 8th International Conference on Knowledge Management in Organizations*. Springer Netherlands, pp. 579–591.

Šmite, D., Wohlin, C., Gorschek, T. and Feldt, R. (2010) 'Empirical evidence in global software engineering: a systematic review', *Empirical Software Engineering*, Vol. 15, No. 1, pp. 91–118. DOI: 10.1007/s10664-009-9123-y.

Soto-Acosta, P., Colomo-Palacios, R. and Popa, S. (2014) 'Web knowledge sharing and its effect on innovation: an empirical investigation in SMEs', *Knowledge Management Research & Practice*, Vol. 12, No. 1, pp. 103–113. DOI: 10.1057/kmrp.2013.31.

Soto-Acosta, P., Popa, S. and Palacios-Marqués, D. (2015) 'E-business, organizational innovation and firm performance in manufacturing SMEs: an empirical study in Spain', *Technological and Economic Development of Economy*. DOI: 10.3846/20294913.2015.1074126.

Treude, C. and Storey, M.-A. (2009) 'How Tagging Helps Bridge the Gap Between Social and Technical Aspects in Software Development', in *Proceedings of the 31st International Conference on Software Engineering*. IEEE Computer Society, pp. 12–22. DOI: 10.1109/ICSE.2009.5070504.

Trkman, P. (2010) 'The critical success factors of business process management', *International Journal of Information Management*, Vol. 30, No. 2, pp. 125–134. DOI: 10.1016/j.ijinfomgt.2009.07.003.

Vera-Baquero, A., Colomo-Palacios, R. and Molloy, O. (2013) 'Business Process Analytics Using a Big Data Approach', *IT Professional*, Vol. 15, No. 6, pp. 29–35. DOI: 10.1109/MITP.2013.60.

Walton, M. (1986) *The Deming Management Method*. Penguin, New York, NY, USA.

Wu, D., Yang, D. and Boehm, B. (2010) 'Finding Success in Rapid Collaborative Requirements Negotiation Using Wiki and Shaper', in *Proceedings of the 43rd Hawaii International Conference on System Sciences*. IEEE Computer Society, pp. 1–10. DOI: 10.1109/HICSS.2010.210.